

# 数据库管理系统软件硬化

刘 甲 耀

〔计算机科学(电脑)系〕

## 提 要

数据库管理系统软件硬化是数据库管理系统及体系结构研制的一个新方向。本文论证数据库管理系统软件硬化的必要性和可能性,综合论述数据库管理系统软件硬化的几种途径和结构型式,指出数据库管理系统软件硬化的发展趋势。

### 一、问题的提出——数据库管理系统软件硬化的必要性和可能性

数据库乃是数据信息之集合。随着社会信息的日益繁杂众多,人们越来越迫切需要用数据库来存贮信息,并用计算机数据库管理系统对它们所需要的大量信息进行处理与管理。数据库管理系统(DBMS)的基本目的就是改善对数据资源的组织控制和使用。当前,DBMS是用软件实现的,它包括模式或数据库定义、数据库本身、子模式或用户模式、以及DBMS过程等四个部分。DBMS检索步骤如图1所示。

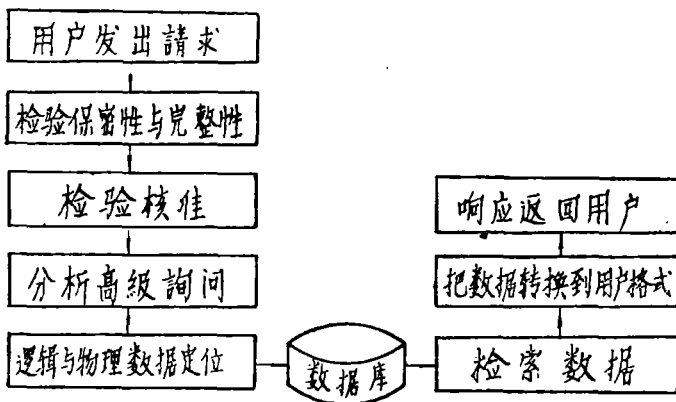


图1 DBMS检索步骤

然后,目前的通用计算机(我们把它叫做常规计算机),由于它是沿用 Von Neumann 为数值计算而计的可编程计算机,数据是用存贮器中的位置来表示的,数据与存贮器的地址间能有效而简单地一一对应,数值运算指令(如加、减移位等)也与机器指令十分接近,这样,数据、运算和机器三者密切配合就能使操作简单而有效地进行数值计算,但用于非数

值计算则不然,因为在非数值应用中,需要用数据内容来寻址而不是用存贮数据的存贮位置来寻址,而且,需要的基本运算也不再是以前的加、减、移位等,而是在存贮器上进行大量的检索更新、插入、删除、以及数据移动等非数值操作。这就使得常规计算机与数据库应用之间产生严重失配。因此,要在常规计算机上实现大量的数据处理的功能,就必须配置庞大的数据库管理系统软件,其中包括应数据库目录,确定存取方法,选择存贮方式等,如要检索所需的信息,就要将目录从辅存调到主存,选取合适的目录索引,再根据目录索引将有关数据调到主存进行检索,找出所需的信息。很显然,这种工作方式有着严重的缺点,即由于大量数据在主存与辅存之间频繁调运,从而使得通道拥挤,CPU繁忙,即使执行一个简单的“删除”、“检索”或“更新”操作也比相当复杂的数值计算还要费事。

另外,随着用户对数据要求的日益增长,数据库应用的日益发展,使得它的使用环境也变得越来越复杂,这也使得数据库管理系统软件变得越来越庞大,越来越复杂。这样庞大而复杂的软件系统,当然容易发生故障,从而降低了系统的可靠性。因此,用目前的常规计算机进行非数值操作,它的性能要比数值计算降低好几个数量级,响应时间也显著增加。近几年来,由于数据库系统的迅猛发展,就使得上述矛盾越来越严重,因而提出了数据库管理系统软件硬化的必要性;另一方面,由于IC技术特别是VLIS的出现和存贮技术行的发展,以及各种微处理机的迅猛发展,这就又提供了数据库管理系统软件硬化的可能性。

因此,人们就设想将数据库管理功能用硬件来实现,即数据库管理系统软件硬化,以改善上述不利因素。这样,就开展了数据库计算机(DBC)的研究。

DBC的关键概念就是从主机去掉全部或部分DBMS,而把它放置在后端机系统上。后端DBC基本上既可以是通用小型计算机,也可以是专为数据库管理设计的专用计算机,DBC只不过是一个常见的子系统,而不是一个完全的计算机系统。采用后端DBC,可以使得:从(1)主机上去掉包括数据库操作、维护、及控制等冗长而费时的运算;(2)通过功能专门化以及主机与后端(DBC)之间平行处理来增加系统的功能。后端机方案当然是为了减少数据管理的费用,也可作为降低价格的有效方法,并能提高主机性能甚至得到常规计算所不能提供的性能与功能水平。后端计算机系统如图2所示。当然,用DBC取代目前正在

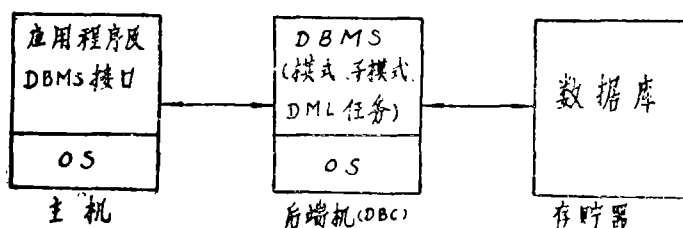


图2 后端计算机系统

运行着的、用软件来实现的数据库管理系统,有两点是要考虑的:一是现有的数据库存贮格式必须变成新的DBC的存贮格式,即所谓数据库变换;二是在现有的所用的数据库语言及子语言均应实时地为新DBC指令所支持,即所谓查询翻译。DBC作为后端机(可以是一台或多台)与一台或多台前端通用计算机以及一个或多个存贮器(数据库)相连接,用户程

序在前端计算机上运行, DBC则作为它们的信息源, 前端机通过数据库命令与 DBC 通讯, DBC 执行命令之后, 或者送回一组记录, 或者送回执行命令成功与否的信息。这样一种工作方式, 使得常规计算机与 DBC 均能获得最佳性能, 也减少了通道拥挤, 改善了响应时间, 解放了 CPU (前端机), 不但提高了数据库管理系统的可靠性, 而且可以大大缩小软件规模, 减少系统开销。

## 二、数据库管理系统软件硬化的途径和结构型式

根据国外对数据库管理系统软件硬化的研究, 可以将 DBMS 硬化的模式归结成五类:

- 1、单处理机间接检索 (SPIS);
- 2、单处理机直接检索 (SPDS);
- 3、多处理机直接检索 (MPDS);
- 4、多处理机间接检索 (MPIS);
- 5、多处理机组合检索 (MPCS)。

### 一) 单处理机间接检索 (SPIS)

单处理机间接检索相当于常规的通用处理机。用这个传统方法, 数据库的一部分是从它的转动磁头磁盘上的永久存贮器读入随机存取存贮器的中间级存贮器。索引表和指示字是用来决定筹划进入系统主存的一部分或许多部分。然后, 处理这个数据块, 以决定检索或修改的记录。虽然, 实施细节可能有所不同, 但所有目前的 DBMS 的操作在本质上讲都是用这种方法。例如, 用索引顺序存取方法, 装入数据库的一部分是由主要的关键值来选定的。在 CODASYL 数据库中, 许多组常常是存贮在它们自己的记录附近, 因此, 得到单数据块增加相似性, 所有需要的记录将在检索数据时找到。

### 二) 单处理机直接检索 (SPDS)

单处理机直接检索是用专用设备来进行检索。目录可寻址的文件存贮器 (CAFS) 就是一种快速检索设备,

而且是单处理机直接检索方法的一个例子。由于 CAFS 不使用数据库模式所提供的附加定义信息, 所以它能用来处理文件和数据库系统。包含 CAFS 的组合检索系统如图 3 所示。

数据库和文件存贮在常规的磁盘上,

为了得到一个比较高的、均匀的数据传送速率, 可以多路转换从几个磁盘输出。从磁盘流出来的数据通过执行选择的小型计算机控制的流程。适合选择标准的记录则传送到主机。最后

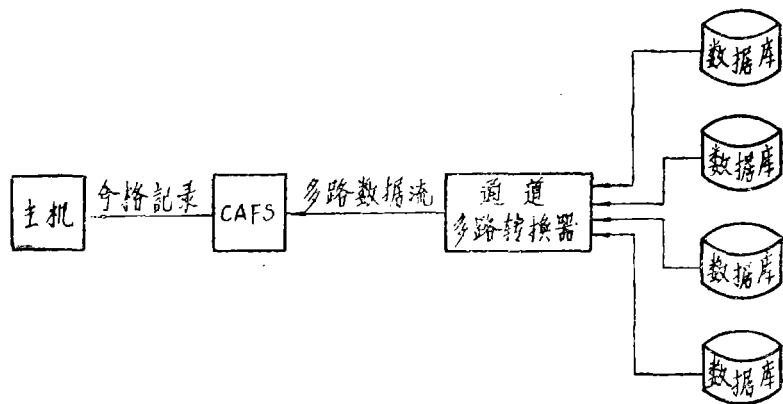


图3 CAFS——组合检索系统

把记录提交给用户而完成处理。

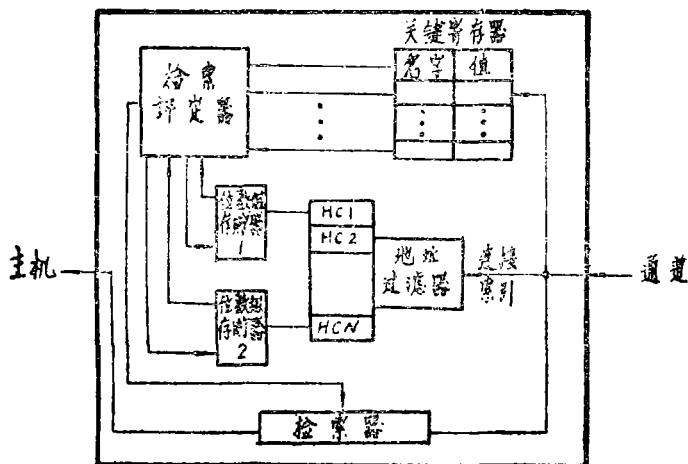


图4 CAFS 系统结构

CAFS 内部体系结构如图 4 所示。16 个关键寄存器是用来比较流经系统的数据。首先，把关键字变量名和值装入关键寄存器，然后由于读记录，同时通过每个关键寄存器行进处理。在每个记录中的标号信息组确保在关键寄存器中的值只与记录中合适的信息组进行比较，也就是说，当标号信息组匹配时，则比较该值。这些比较的结果确定三个组合门（小于、等于和大于门）之中的一个门置位。这些结果也传送到检索评定器，以评定询问的完全表达式。为了连接和消除复制后面的投影，多重 64K 位数组存储器是组合在 CAFS 中，并能用来保存检索评定器的输入数据。最后，检索器在它的缓冲器中收容记录，直到记录核定为止。如果记录是合格的，则检索评定器把开关置位使记录来传送到主机。否则，在读下一个记录时，记录就被取代或重写到检索器。

### 三) 多处理机直接检索 (MPDS)

多处理机直接检索方法的基本特征如图5所示。利用处理机每个磁道来直接检索数据

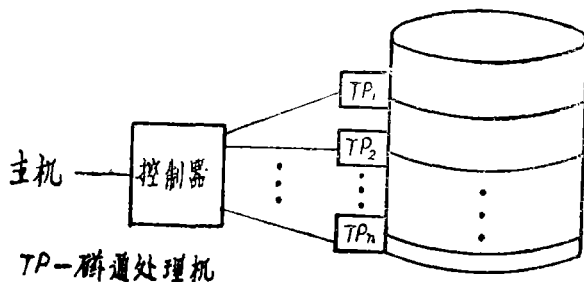


图5 多处理机直接检索

库。由于存贮设备（例如磁盘，磁泡，或者电荷耦合器）旋转，数据就读入相应的磁道处理机，然后，读处理机就检查记录以确定应该选定或修改的记录。用MDS，由于检索直接在存贮设备上进行，减少了通过系统的数据容量。同时由于数据库每个磁道有一个头部，检索

时间实质上与数据库大小无关，并可以在一个单旋转中检索整个数据库。实际上，在单旋转中用多重比较寄存器就可能回答若干请求。上下文可寻址的程序段顺序存贮器（CASSM）就是处理机直接检索的一种设备。它是专为通用的非数值应用（如数据管理，信息检索，以及文件处理）而设计的。在数据库上下里面，CASSM 允许对数据库进行并行处理，因此，执行许多数据管理功能所需要的时间与数据库大小无关。CASSM系统的结构如图6所示。它

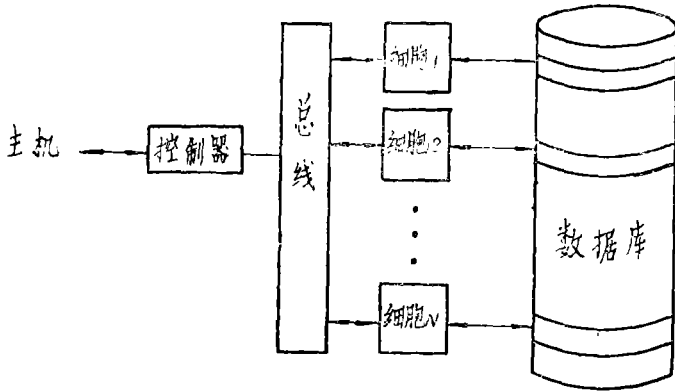


图6 CASSM 系统结构

由存贮器和处理机两部分组成。在这个系统中被处理的文件是分区的，而在每个细胞中放置的部分信息可以并行处理。能够用这种方法存贮的数据库的容量是许多细胞和每个细胞的存贮容量的函数。整个数据库可以在磁盘旋转中处理。如果数据库大于所有 CASS M细胞提供的 存贮容量，则它必须是分区的，而且分别进入可变存贮器。CASSM 细胞结构如图7所示。存贮器可以用固定的磁头磁盘，磁泡，电荷耦合器等设备来执行。由于数据循环，每个字是读，处理，以及写回到这个存贮设备中。

与常规的存贮旋转不同，CASSM 细胞含有单独的读头和写头。而且每个存贮字可以读，如果不删除它的话，则在每次旋

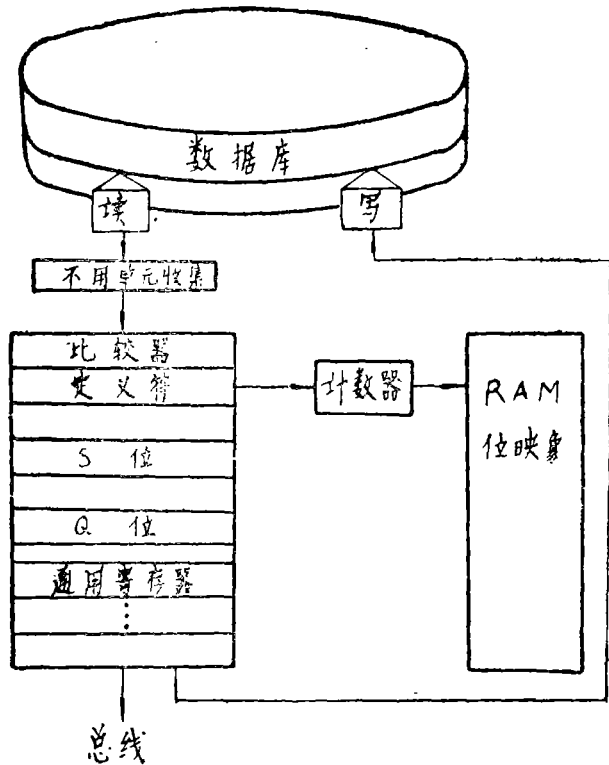


图7 CASSM 细胞结构

转时写回。插入和删除情况如图8所示,是五字传递途径的例子。首先,由存贮器读入传递途径的第一个字,在传递途径中的每个字则下移一个位置,第5个和最后的字则移出传递途径并丢失。为对每个字读和移入传递途径,则写磁头必须选定一个字,并把它写回到存贮设备中。正常情况写出传递途径中间(位置3)的字。如果该字是特征位,则作为不用单元的字删除,然而,下一个字(目前在位置2)则被选中写出。这个旋转的其余部分,

所有被写出的字均在第二位置选定。这个过程允许一个不用单元字在每次旋转时从存贮器上消除。在这个例子中,通过执行同样的操作就能在单旋转中删除两个字,并从位置1写出其余的字。

相反的操作能用来在数据库中插入字,取代位置3写出的字,写出所插入的字,然而,老字不丢失,因为从传递途径输出是移位的,而且是从位置4选择的。其次,对于旋转的其余部分,字是从位置4写出,而不是从正常的位置3写出。正如此例一样,每一旋转能删除两个字,插入两个字。

#### 四) 多处理间接检索 (MPI3)

多处理机间接检索类似于多处理机用来并行处理数据库的情况。所不同的是,整个数据库不是用平行处理,而是数据库的一部分放到中间的存贮设备上,并在那里进行检索。多处理机间接检索方法如图9所示。这种方法是有生命力的。这里需要一种途径来快速识别需要

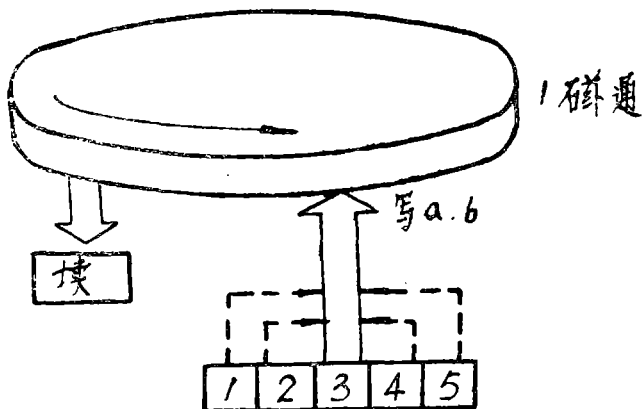


图8 CASSM 插入或删除

a—正常的从位置3通过,任何位置是可能的  
b—号码较小的字

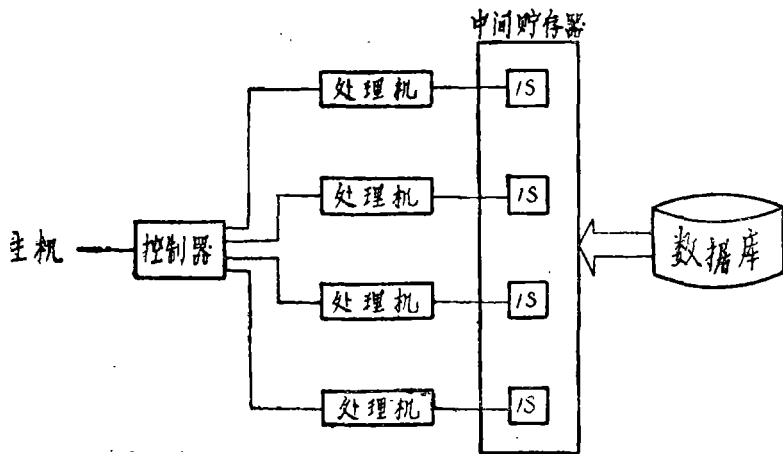


图9 多处理机间接检索

处理的数据库的那一部分，并把它们输入到中间存贮器。模型条件需要指示字或数据库的专门部分的索引。这些指示字能包括大数据块而不是个别的记录。后一条件，快速装入中间存贮器要求中间存贮器里均具有高数据库传送速率和缓冲。利用平行转换磁盘的新方法在转动的磁头磁盘上读写多重磁道，，有可能及到比较高的传送速率。

用 MPIS 方法，处理事务的时间至少是间接地取决于数据库的容量。就需要检索的数据库的部分来说，将适合在中间存贮器里面，而处理则与数据库容量无关。换句话说，中间存贮器装入和处理需要若干时间，因此，响应时间将随数据库容量而增加（或者至少是成比例关系）。这种增加不是线性的，然而，在中间存贮器装满之前什么也没有变化。

利用 MPIS 的结构有 STARAN 和 RAP 两种。

### 1、STARAN 结构

第一个 MPIS 数据库计算机 STARN 结构起初是为图象处理设计的。STARAN 体系结构如图 10 所示。其中，顺序控器（PDP-11）控制系统内部操作，並管理与主机通讯。

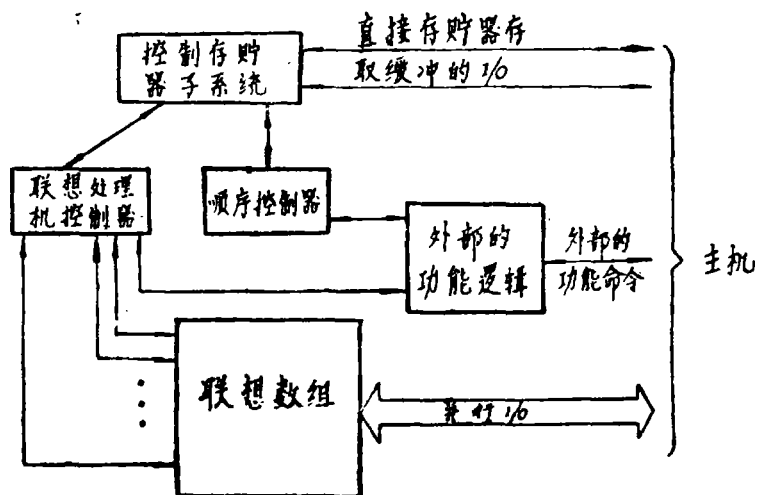


图10 STARAN 结构

一个单微编码联想处理机控制器数组存贮和联想硬件的操作。因为这里只是一组指令逻辑，控制和寻址寄存器，所有数组存贮必须在封锁步（lock-step）操作，即在相同的时间里在它们之中都执行相同的指令。在这方面只允许除封锁步操作之外，当执行具体指令时，如果字选择位置不一，则相应的字是不处理的。单比较和单标记寄存器是供所有数组共同使用，因此，对所有数组提供单比较平行执操作。任住几个关系操作（等于，不等，小于，大于或等于，等等）均能使用这些比较。同时，在检索能够开始之前必须装入所有中间存贮器。因此，装入这个联想寄存器所要求的带宽和时间是系统性能的关键问题。

### 2、RAP 结构

RAP（关系联想处理机）由于不是一个孤立的系统，要求通用主机去编译高级用户询问进入 RAP 命令，调度 RAP 共行操作，并把 RAP 指令传送到控制器。主机同时对所有

的数据库完整性和安全性, 维护关系和域名表以及编码/译码表负责。RAP本身包含有控制器, 组合功能部件 (SEU), 以及许多细胞, 其中每部分均与它的两个邻接的部件连接。RAP 结构如图 11 所示。单个细胞结构如图 12 所示, 包括一个信息检索与操作部件 (ISMU), 一个算术和逻辑部件 (ALU), 一个缓冲器, 以及一个存贮器磁道。每个细胞直接在它自己的存贮器上或者间接在其它细胞的存贮器上, 通过与它的邻近细胞通讯, 执行许多关系操作。信息检索和操作部件 (ISMU) 控制细胞操作, 并提供与系统其它部分的所有通讯。它也包含很少几个 (由硬件设计限制决字) 选择值, 以允许细胞同时处理几个检

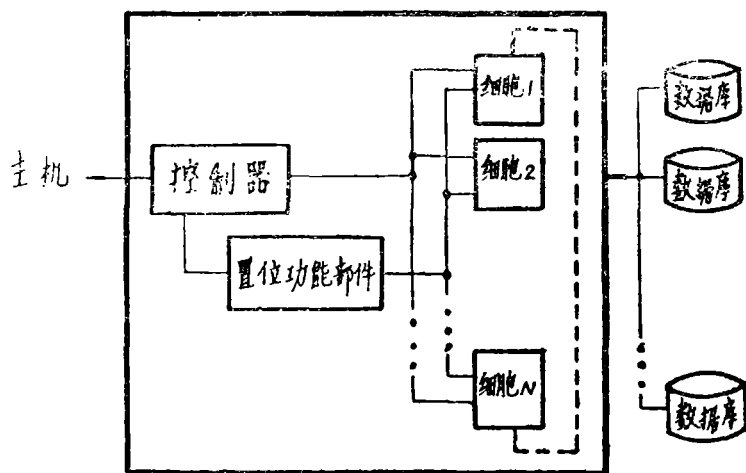


图11 RAP 结构

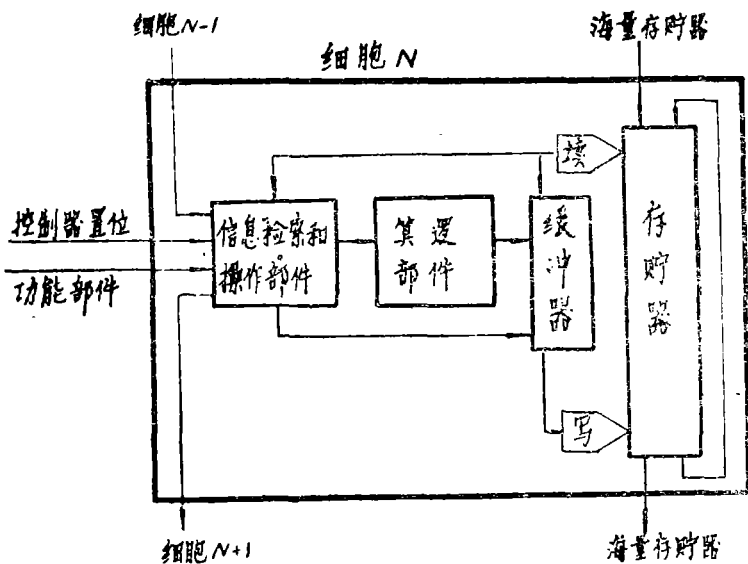


图12 RAP 细胞结构



索。允许细胞同时处理几个检索。因为每个细胞包含自己的比较寄存器，不需要封锁步操作。

在操作上，联想存贮是在每次旋转时读进缓冲器。如果记录符合检索标准，在数据块中适合的标记位就置 1，并且写出记录。随后的旋转，所有标记的记录能够检索并发送到主机。

如果数据是更换的或标术修改的，则在缓冲器中执行操作，然后，写出在新值中修改的记录。如果记录删除的，则删除标记位置 1。在后的时间里，随后的记录往上移动，而删除记录则重写。因而不需单元收集是自动的，在联想存贮器中具有的自由空间聚集在每个磁道的末端。

控制器接收和译码命令来自主机，装入检索标准进入 ISMU 的每个细胞，并控制执行 RAP 命令。

#### 五)多处理机组合检索 (MPCS)

多处理机组合检索方法兼直接和间接检索方法的许多最好的特性。用组合检索方法，检索是在数据装入到中间存贮器执行。同时，把多处理机分配给中间存贮块，而不是直接分配给海量存贮器的每个磁道。因此，系统的费用并不随数据库容量增大而逐步增加。

支持多层数据模型 (包括层次、网络和关系模型) 的 DBC 结构如图 13 所示。DBC 包括七个部件两个循环：一个是结构循环，一个是数据循环。结构循环的功能是在处理时把选择表达式转换成海量存贮器的数据块检索物理地址。在海量存贮器中，这些数据块或最小的可

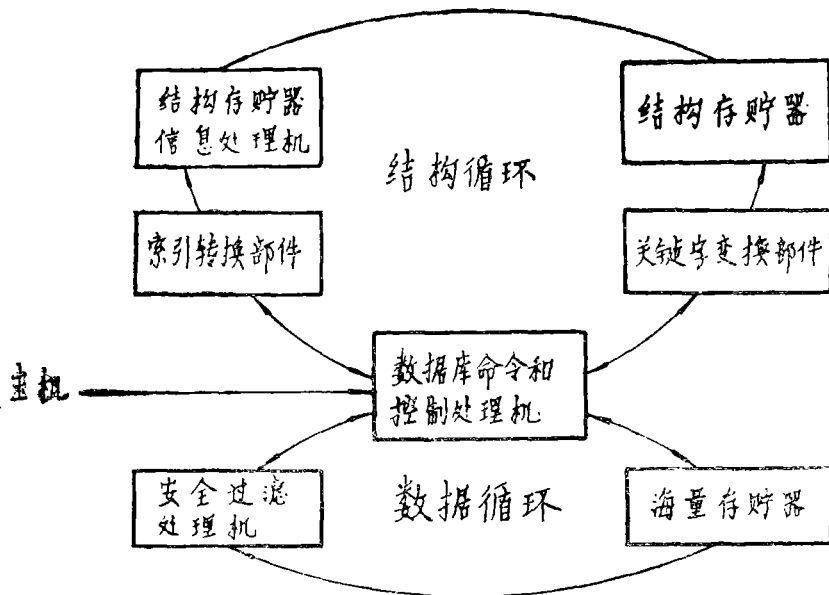


图13 DBC 结构

寻址部件 (MAU) 包括相应于圆筒在转动磁头磁盘上的实际数据记录。在结构循环中四个部件是关键字变换部件 (KXU)，结构存贮器 (SM)，结构存贮器信息处理机 (SMIP)，以及索引转换部件 (IXU)。关键字变换部件把请求转换成许多索引进入 SM。结构存贮

器,数据库的实际转换表,产生数据要求的逻辑地址去回答请求。结构存贮器信息处理机使用结构存贮器的输出来在这些逻辑地址上执行逻辑操作。索引转换部件把逻辑地址转换成物理地址。

#### 用户请求

对 DBC 请求包括三部分:操作码,很少几个关键字属性,检索或更新修改的数据项。关键字属性定义有关的记录,是逻辑表达式,而本身或者为真或者为假。它们包括关键字变量名,关系运算符(小于,等于,大于,或等于,等等),以及值。

#### 数据库命令控制处理机(DBCCP)

在 DBC 中心是数据库命令控制处理机,它接收从主机来的用户请求,并把它转换成很少几个 DBC 操作。然后, DBCCP 控制使用其它 DBC 中间的各个部件来处理用户请求,最后, DBCCP 把请求的数据传送到主机或者进行更新。

#### 关键字转换部件(KXU)

关键字转换部件把单关键字属性转换成结构存贮器可接受的一种形式。在这种形式中(文件名,关系运算符,信息组名)关键字属性虽然不能直接在 DBC 上进行处理,但必须首先分析进入的许多基本关键字属性。所有关键字部分属性(文件名,信息组名,值)除运算符外,可以是可变长度。然而,所有表键字属性部分通过一组变换表转换成定长编码。

#### 结构存贮器(SM)

由 KXU 产生的定长编码是送到 SM 或数据库变换表。SM 的每一个项目包括一个定长代号,几个逻辑实体,而每个逻辑实体则包含一个逻辑指示字,分类标识符,安全性技术要求。SM 是联想检索来找到相应于从 KXU 发来的代码的实体。

#### 结构存贮信息处理机(SMIP)

单独的关键字属性是由 KXU 变换,然后通过 SM 处理的。SMIP 执行逻辑操作,并在关键字属性上组合。对于逻辑或的情况,每个数据块包含可以满足必须检索询问的记录。因为同一数据库可能是用两个关键字属性来标识,然而, SMIP 能够容易地删除付本而不再次检索数据块。SMIP 的另一个重要功能是执行比较困难的逻辑操作。

#### 索引转换部件(IXU)

IXU 接收来自 SMIP 的标识符,并把它们从逻辑指示字转换到物理指示字。

#### 海量存贮器(MM)

DBCCP 对海量存贮器发出一个请求以访问由 IXU 指定的磁盘园筒或最小可寻址部件,关键字属性发送到 MM, MM 必须执行数据的实际测试,以确定合规的记录。

#### 安全过滤处理机(SFP)

SFP 提供比 SM 规定的安全微量更精确的安全性。因为 SFP 检查所选择的实际记录,安全性选择能够在数据项和数据值级执行。SFP 的另一个功能是把记录分类进入由用户规定的序列。实际上, SFP 接收一个平行的记录流(来自各个磁盘磁头),并把它们转换到串行流。在这点上, SFP 能把记录转换到 DBCCP。

### 三、结 论

1、用现行的 LSI 模块或者微处理机来组成 DBC，以取代常规的数据库管理系统软件，联机 I/O 例行程序，以及联机辅助存储器，它们所提供的性能远远超过软件数据库系统，无论是价格/性能，关系数据模型的实际实现，或者数据库的完整性、安全性、可用性以及可维护性等各方面都会大大改善。显然，数据库管理系统软件硬化乃是 DBMS 研制的一个新方向，它有着强大的生命力，DBC 的出现标志着计算技术发展史上一个重要的转折点，意味着用常规计算机进行非数值数据处理的传统方法的转移，预示着用专用后端机 (DBC) 进行非数值数据处理的到来。

2、DBC 在设计方法上把硬件、软件作为一个整体来处理，采用若干微处理机来实现各部件的特殊功能，用时采用高级数据库命令与前端机通讯，支持不同类型的数据库模型，从而使得结构布局合理。以微处理机为基础组成的 DBC 结构可用如图14所示的方案。这是 DBC 的一种扩充，有六个主要部件：

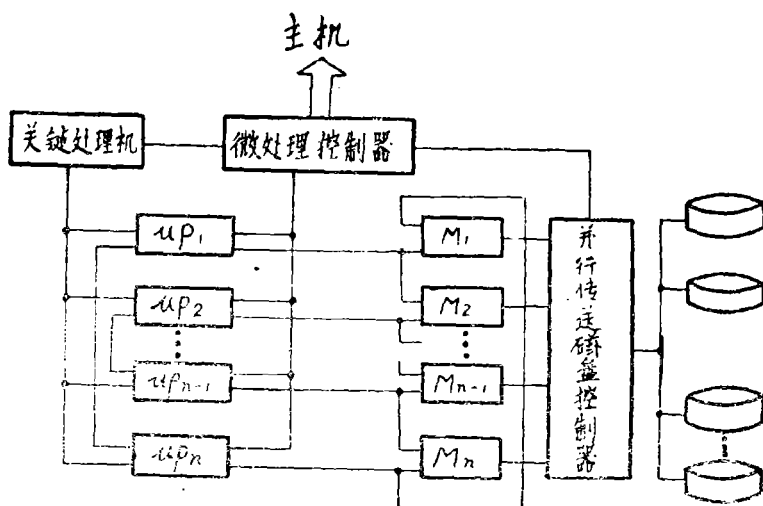


图14 微处理机为基础的 DBC

- 1、微处理机控制器 ( $\mu c$ )
- 2、微处理机处理部件 ( $\mu P'S$ )
- 3、在贮器模块 ( $M'S$ )
- 4、关键处理机 (KP)
- 5、并行传送磁盘控制器 (PTDC)
- 6、并行传送磁 (PTDs)。

微处理机控制器或者接受网络数据操作命令，或者接受关系型或网型的高级询问类型命

令, 然后, 各命令由 $\mu c$ 处理, 而合适的参数和命令是为 $\mu P, S, KP$ , 以及/或者 PDTC 操作产生的。

PTDC 提供常见的错误校正, 故障处理, 以及平常在目前磁盘控制器上发现的其它特性。

关键处理机(KP)是用来加速可靠的数据库操作, 可以用VLSI 技术设计。

来自适合平行传送磁盘的信息同时传送到有关的缓冲器(即存贮模块)。给定的存贮模块有两个存贮体, 以便交叉执行输入/输出操作。

奇偶数的 $\mu P$ 's 分别与总线机构连接。这些微处理机的互连以及与邻近的存贮模块连接, 允许做诸如选择, 投影、完全连接、隐式连接、附加、修改、删除和分类等操作。根据这些操作得到的数据接着返回到主机或送至磁盘上。

### 参 考 文 献

- [1] Bray, Olin H., Data Base Computers, D.C.Heath and Company, 1979.
- [2] Gio Wiederholz, Database Design, McGRAW—HILL Book COMPANY, 1977.
- [3] A. T. F. Hult, A Relational Data Base management System, JOHN WILLY and SON, 1979.
- [4] G. Bracchi and G.M. Nijssen, Data Base Architecture, 1979.
- [5] C.J. DATE, An Introduction to Database Systems Second Edition, ADDISON—WESLEY PUBLISHING COMPANY, 1979.